CORE
Metadata, citation and similar papers at core.ac.uk
Provided by AIS Electronic Library (AISeL)

Scandinavian Journal of Information Systems

Volume 9 | Issue 1 Article 5

1997

DARWIN: Message Pad Support for Networked, Dispersed Groups

Steinar Kristoffersen
Norwegian Computing Center, steinar.kristoffersen@nr.no

Fredrik Ljungberg
Viktoria Research Institute, fredrik@informatics.gu.se

Follow this and additional works at: http://aisel.aisnet.org/sjis

Recommended Citation
Kristoffersen, Steinar and Ljungberg, Fredrik (1997) "DARWIN: Message Pad Support for Networked, Dispersed Groups," *Scandinavian Journal of Information Systems*: Vol. 9 : Iss. 1 , Article 5.
Available at: http://aisel.aisnet.org/sjis/vol9/iss1/5

This material is brought to you by the Journals at AIS Electronic Library (AISeL). It has been accepted for inclusion in Scandinavian Journal of Information Systems by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

DARWIN: Message Pad Support for Networked, Dispersed Groups

Steinar Kristoffersen[1] & Fredrik Ljungberg[2]

[1]*Norwegian Computing Center*
P.O. Box 114 Blindern, N-0314 Oslo, Norway
steinar.kristoffersen@nr.no

[2]*Viktoria Research Institute*
Viktoriagatan 13, S-411 80 Gothenburg, Sweden
fredrik@informatics.gu.se

Abstract

Our empirical work implies that mobile, distributed and networking groups often experience problems with regards to sharing experiences and coordinating work effectively. In this paper, we ask: How can we facilitate sharing and coordination in the dispersed, mobile and networked group by means of IT? *Our answer for a specific case is an application called DARWIN. The design of DARWIN has been informed by ethnographic work, qualitative interviews, and various evaluation sessions. Describing DARWIN and the process that led to it, serves an illustration of a more general answer to the research question asked.*

1. Introduction

In this paper we describe and reflect on the design of DARWIN (Direct Access to Resources, Work & Information in the Network), a message pad application supporting cooperative work in an information technology (IT) support group. DARWIN manages two *common information spaces* (Bannon & Kuutti 1996, Bannon & Bødker 1997) to facilitate coordination and sharing in a dispersed, mobile, and networked setting.

The work in the support group is networked, dispersed and mobile. Being *networked* (see, Pfeffer 1982, Thompson & Frances 1991), rather than bureaucratized, the group members are not usually assigned tasks by their manager, nor are they generally encouraged to perform their work in a particular way, or engage in collaboration with certain persons

© Scandinavian Journal of Information Systems, 1997, 9(1):3–24

within the local organization. The group staff usually take on tasks according to personal interests and the perceived needs of the users, planning and articulating their work locally, relying heavily on personal contacts rather than on group functions for commissioning and work support. Another characteristic aspect of the networked support group is that they are geographically *dispersed*. Being co-located with the user communities is vital in providing close, continuous, and efficient support services to promote effective IT use in the company. Providing support services in the dispersed and networked setting, the group members are highly *mobile*; walking rounds to be visible and visit the users, etc., make mobility an integrated and essential part of the work in the group.

Our empirical work implies that the group experiences problems in sharing experiences and coordinating work effectively (see, Kristoffersen & Ljungberg 1996). Exploring these problems in more detail, we concluded that they could not easily be addressed by introducing other organizational structures (Kristoffersen & Ljungberg 1996). For example, we found that it is vital for the group members to be co-located with their user community to be able to provide effective support services. Likewise, the personal networks and the mobile nature of work turned out to be essential for offering close and proactive IT support. It was therefore natural to ask how the perceived problems could be addressed by means of IT. The research question thus asked was:

> How can we facilitate sharing and coordination in the dispersed, mobile and networked group by means of IT?

The suggested answer, presented in this paper, is DARWIN. DARWIN consists of three parts:

- A shared experience base containing experiences gained in the group.
- A common task space supporting the co-ordination of work.
- Newton PDAs (Personal Digital Assistant) for access from mobile settings.

The paper is organized as follows. Section 2 outlines related research, followed by the research background. Section 4 describes DARWIN, while section 5 presents the evaluation of the application. Section 6 discusses the use of DARWIN, and section 7, finally, concludes the paper.

2. Related research

A significant body of research exists on how to create and share knowledge within groups and organizations (see, March 1991, Cook & Yanow 1993, Kim 1993, Bannon & Kuutti 1996). Concepts such as "common information space" (Schmidt & Bannon 1992, Bannon & Bødker 1997) and "organizational memory information systems" (Ackerman 1994, Stein & Zwass 1995) have been invented to describe IT supporting such processes.

Current CSCW systems go about the process of recording information in a number of ways. QuestMap, previously called gIBIS and CM/1, is one of the better known systems that aim to support memory in organizations. It implicitly takes on board a concern to record only the rational aspect of decision making, thus limiting, by its selection and struc-

turing of information, the number of organizational situations in which this information might later become useful (Conklin & Begeman 1988, Yakemovic & Conklin 1990, Conklin 1993).

Answer Garden is a systematic attempt to augment the expertise of an organization, motivated by "an impetus from layoffs, down-sizing, and internationalization of personnel" (Ackerman & Malone 1990). The system is concerned with making recorded knowledge and live experts available for the users. It allows users to look for answers to diagnostic questions that form a tree structure. If a satisfying answer is not found, the user is offered to ask the question to a designated local or global expert. The questions are mediated by email. If the expert chooses to answer, he does so by email. If the question is incomprehensible, inconsistent, or incomplete, the expert can ask the user to elaborate, and vice versa on the answers. The mail systems are initiated from Answer Garden, but the continued conversation and the replies are detached from it. To the extent that the expert updates the database after dealing with a query, this constitutes the iterative construction of the corpus of information. A similarly strict organization of information is enforced by TeamBuilder (Karduck 1994). TeamBuilder is based on a hypertext-like, traversable network structure, where experts, teams, tasks, and projects are represented as objects in the system architecture. Answer Garden and TeamBuilder would not be suitable solutions for our purposes, primarily because they do not run on mobile computing devices.

Research on co-ordination of work has, amongst others, been carried out in the COMIC project.[1] The notion of "Mechanisms of Interaction," (Schmidt 1994b, Sørensen *et al.* 1994) later evolving into "Co-ordination Mechanisms" (Schmidt & Simone 1996), is a well-known approach to supporting division and co-ordination of work by means of IT. Similar approaches have been discussed by, amongst others, Migliarese and Paolucci (1995), Holt (1988), Malone and Crowston (1992), and Winograd and colleagues (Winograd & Flores 1986, Medina-Mora *et al.* 1992).

However, there has been no research on coordination and sharing in dispersed and mobile groups. Accordingly, the particular conditions for mobile service workers, such as the IT support group, has not been addressed in CSCW previously. Some previous work has been carried out on mobile work, listing specific requirements for information systems and applications (Kristoffersen & Rodden 1996). Bellotti and Bly (1996) found that whilst much work is significantly mobile, there is very little support in terms of applications available. On the platform level, however, some commercial and research toolkits are available: Lotus Notes can replicate document databases, *but* it does not include a mobile client.

Coda supports disconnected operations from client workstations (Kistler & Sayanarayanan 1992), but application development is cumbersome. Other platforms, for instance Bayou (Terry 1995) from Xerox, Rover (Joseph *et al.* 1997) from MIT and Sync (Munson & Dewan 1997) from Xerox, Rover from MIT and Sync from the University of North Carolina, are concerned with the architectural issues of mobile computing, such as synchronization and conflict management. The main reason why Coda and Rover

would not be appropriate for our project is that neither of them cope effectively with consistency. The main problem with Bayou is that it is not suited for the object oriented databases needed in our application.

Service work is generally being done closely interacting with the customer (e.g., Svensson & Orban 1995). Often, this makes services highly situated and dependent on factors that are difficult to predict in advance (Ljungberg 1997). Therefore, ad hoc interaction is likely to be frequent (Ljungberg 1997). This warrants flexibility and openness of coordination mechanisms in these kinds of settings.

To summarize, even though there is much literature on coordination and sharing, the particular setting explored in this paper, i.e., the dispersed, mobile, and networked service group, has *not* been covered previously. This is the main motivation for the work presented in this paper.

3. Research background

The following section introduces the background for the research presented. A brief description of our research approach precedes the requirements and design considerations elicited from the fieldwork.

3.1. The Setting

The pharmaceutical research company in which we conducted this study employs about 1000 people. It is a significant player in the European market, with a $400,000 turnover per employee. About 750 employees are directly involved in the research, spanning from basic cell biology to innovation in pharmaceutical chemistry. Our fieldwork focused on the "Clinical Information and Data management, Education and Support" group (CIDES), the clinical division's IT support group. Organizationally, the group belongs to the "Information Technology & Data Management" department (IT & DM). The clinical division, employing about 325 people, advances research on drugs that have already passed through the pre-clinical research, by evaluating them on human subjects in collaboration with clinics all over the world.

The IT support group consists of nine people; six persons are directly involved in providing support services, one system administrator, one secretary, and one manager. The six members working on direct support are responsible for one or two departments in the clinical division. One important aspect of the organization of their work is that the support group members are co-located with their users, i.e., they are geographically dispersed. The manager of the support group, the secretary, the system administrator and one of the support people, are located in the same building. The other five support staff are dispersed at different floors and annexes in the company's main building. The distance between the two buildings is about 200 meters, crossing a busy through-road.

3.2. Method

Our research objective involves drawing design implications from a real-world working situation. We therefore chose to do a focused qualitative study. Combining participant observation, interviews and a group discussion, we believe that we gained a sufficiently accurate view of how work was accomplished in the sup-

port group. Approximately 75 person-hours of ethnography (see, Hammersley & Atkinson 1993) were conducted, individually by each of the two authors. Everybody in the support group was aware of our role as observers and the purpose of the study. We continued the fieldwork by interviewing all members of the group, except the secretary, since we had the opportunity to talk to her about her work quite extensively during the observation. An interview guide approach (Patton 1990) was applied, and the interviews, lasting for approximately one hour, were taped.

A group evaluation session followed the interviews. The main purpose of the evaluation session was, first, to validate our suggestions of new IT use, and second, to facilitate the group members reflecting upon the possibilities of future IT use in their work. Since the empirical study, we have been engaged in other informal design briefings with individual group members, most importantly a meeting with the group's human factors specialist.

3.3. The work in the network

Members of the IT support group respond to a wide variety of requests. It would be impossible for one person to know how to solve all possible problems. Not all members are allowed the privileges to do any set of operations potentially implicated by the tasks. For tasks that imply operations outside their domain, the support staff deal extensively with people external to the group. The implication for the support group members is that often they cannot immediately conclude a commission; they are interdependent on other members of the organization at large (Schmidt & Bannon 1992, Schmidt & Simone 1995), not only to have a job, but also in getting the job done.

Being immersed in their area of responsibility, typically one or two adjacent departments, the people in the IT support group get to know their customers and the ways in which they do their job. The knowledge of the work arrangements enables the members of the IT support group not only to perform reactive support, which is the common way of most IT support groups, but also what they call proactive support. Reactive support takes place when the users experience problems in the use of IT and contact their support person, or the central help desk, to get help solving these problems. Based on the combined insights of how people work, and specialized competencies in IT, proactive support concerns suggesting how people could use IT more efficiently in their work. As indicated by recent research (Bowers *et al.* 1995, Sachs 1995), such work requires a thorough understanding of how work is actually performed, which is the most important reason for the support group, opposed to an ordinary helpdesk, to be dispersed in the organization.

The characteristics of work in the support group do not straightforwardly conform with traditional group work (see, Ciborra 1993) which often has been described as having a more regular nature, e.g., with pre-defined procedures for division and co-ordination of work (see, Schmidt 1994a, Carstensen *et al.* 1995). At the same time, however, the support dispersed and mobile group would clearly benefit from traditional group functions. Due to an unpredictable and changing workload, the members of the support group found it difficult to

partake in design projects improving or designing new applications for the researchers. Project participation (25% is the goal of the group) is considered as a way in which the competencies of the support staff can be recycled into IT design. Partaking in projects is also thought of as a way of inspiring and educating the group. Currently, however, members of the group never know, due to the way work is organized, if they can make it to the next meeting. Another aspect is that of specialization. Sometimes very specialized skills are required, e.g., to maintain a Lotus Notes database, install new applications on a Newton message pad, or prepare a data file for statistical analysis. In a group, people can cover for each other to create free time in which people can specialize, and mechanisms to take advantage of someone else's competencies. In a networked, distributed organization this has turned out to be very problematic.

3.4. Requirements

The fieldwork from the research company is presented in more detail in (Kristoffersen & Ljungberg 1996), along with a general discussion of how actors in a networked organization of work could access the group support functions using IT.

Currently, the members of the support group enter tasks requested from the users, as they are made, in a notebook. If they want other people to help them perform a particular task, or wish to make their experiences available to the group, the chore of rewriting and elaborating the notes is inevitable. It must be possible to access the information outside the office, simply because most of the work happens elsewhere. The support staff are commissioned, and hence require IT support, in a variety of locations. For example, in their users' offices when encountering people in the corridor, meeting customers at the coffee machine, and so on. It must be possible to update the common information spaces independently of localization. Many tasks are concluded even before the group member can access his own workstation.

We suggest using a digital message pad to replace the pen and paper journal. A premise for the introduction of a handheld computer is that data entry should be at least as easy as using pen and paper (see, Preece 1994).

Important requirements elicited from the fieldwork are:

- People can only be expected to enter information once (see, Grudin 1994), and we therefore suggest that all shared information would have to be entered directly into a computer supported repository, or information space. The group have twice tried to share work notes previously. Both applications were used initially, but soon failed due to lack of interest. Members told us they were required to enter the same information twice: first when they did the work, and later upon entering it into the database. Generally, few incentives exist to document one's activities for the benefit of others (Grudin 1994). Information for a shared repository should thus have to be entered electronically as part of a required and existing activity in the current organization of work.
- Due to the mobile nature of the support work, a variable, but often very high, workload, and the fact that task

descriptions and documentation often have to be entered whilst on the phone or already engaged with another user, the members should only be required to enter a minimum of information. Several strategies could be combined to comply with this requirement, e.g., semi or fully-structured input formats with fixed semantics, and menus or radio buttons being the preferred alternative to freehand writing.

- Because IT support depends on the work of others, tasks cannot be expected to be completed in any specific sequence. Members are faced with a constantly changing workload and interruptions are common. New IT must hence not assume that tasks are accomplished in a certain sequence or that one task is completed before another is begun (see, Bowers *et al.* 1995, Sachs 1995).

We suggested for the support group to implement two common information spaces: *one shared experience base* containing task-related notes collected by the support group members, and *one common task space* supporting the co-ordination of work. Furthermore, a Newton message pad should be used for each member to *facilitate mobile use* and access to the shared databases.

The shared experience base should contain experiences recorded by the support staff during their work, e.g., how to solve a particular problem, who to deal with regarding new hardware, when to expect network maintenance starting next weekend, etc. The common task space could also be used to offer and request the assistance of peers, by sharing work description items across the network.

Information for the shared spaces should include its originator, for the following reasons:

- Since a minimum of information will be entered, it might be difficult to interpret the situation correctly.
- It is hard to express "lessons learned" briefly and at the same time clearly.
- The data could be hard to reuse correctly, even if the account was complete (see, Allan 1977, Schmidt & Bannon 1992, Langefors 1995).

The requirements listed above has been realized in the DARWIN application. Making it possible for the members to contact each other sustains mobile information spaces for co-operative work.

4. DARWIN

In this section we present the design of DARWIN, explaining which features it involves and our suggestions for how it can be used in the work of the IT support group.

4.1. Entering tasks

The data entry operation of DARWIN provides support for entering new tasks, as well as updating and keeping track of the tasks already initiated.

To replace the manual notepads that the support staff currently use to keep track of the status of initiated tasks, the main design requirement for the data entry operation is that it should be at least as easy to use as the manual system. Responding to this requirement, we proposed a minimal set of optional fields on

the message pads data entry view. The fields we propose are commonly used in the manual journals people currently use.

If the support people cannot instantly deliver the service requested of them, they typically want to enter the name of the customer, i.e., *who* wants something done. Other issues to be entered are: the *priority* of the problem (ranging from low to very high), and who initially will be *responsible* for the task. It should be noted that tasks are not automatically transferred between people, but must be proceeded by negotiation. In the current version of DARWIN, the support staff have to enter "what" and "who" for all new tasks. In future versions a selection menu approach will also be offered as an option.

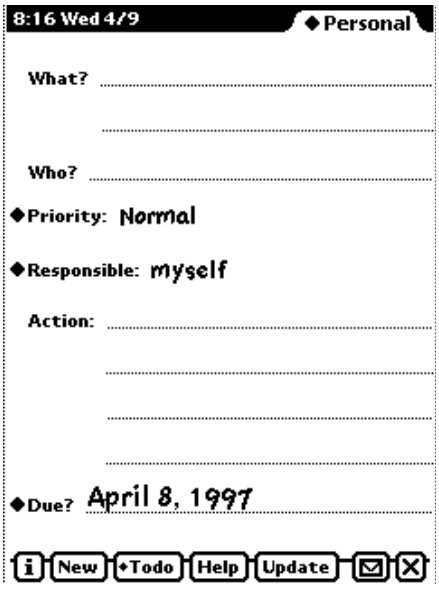
A selection menu is currently offered for the "responsible" field, i.e., who the support person believe would be feasible to accomplish the task. We also consider the possibilities of adding a "keywords" option and thesaurus features, to make it easier for the support staff to classify new tasks. A more homogenous use of concepts within the group could, arguably, improve the possibilities for effective co-operation. This is due to the fact that the support staff are often forced to write down the customers problems and contact someone else for assistance. These contacts might, thus, be carried out more effectively and smoothly by using a similar vocabulary.

The next field describes the *Actions* (to be) taken in order to resolve the problem. In this field the support people are assumed to enter a brief description of how they resolved the particular task, but only inasmuch as they believe that is valuable information for their future work. Writing down the actions taken in this way should be the most important way of making experiences persistent and possible to share within the group.

Since the message pad is hardly optimal for complicated editing of elaborate texts, we found it important to enable staff to update the shared information spaces from a desktop PC. Most of the time, however, that will not be an issue.

The priority menu offers the user to specify the relative importance of requests. The handling of the record in the common information spaces relies heavily on this attribute, as we will show in Figure 1.

FIGURE 1. The data entry view in DARWIN



Beneath the data entry part of the view is the application toolbar, in which actions are represented by folders. By tapping the desired icon, the screen

changes, but the status information is stored so that the user can return easily to the data entry view.

4.2. Searching the common experience space

DARWIN supports the process of sharing experiences amongst the support people through the management of a local instance of the common experience space, consisting of successfully completed tasks by the members of the support group.

By helping the group to share experiences, we believe that the amount of work to "reinvent the wheel," which appears to be quite common today, will be reduced. When users press the *Help* icon, this feature of DARWIN takes the text entered in the record as an argument to a query searching for similar records in the experience base. The significant attributes of the record in this situation is (see Figure 2):

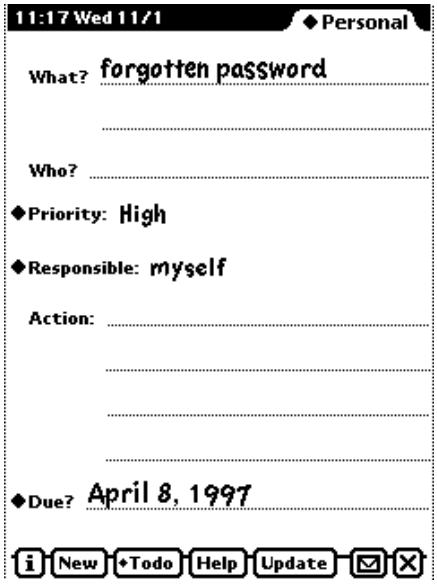
- *Who* did the work, in case the member will need to make contact to get additional information about how the problem was resolved. This is partly due to the fact that DARWIN is designed to require the user to enter a minimum of information, which might make the process of interpretation harder (Langefors 1995). Another reason for recording who did the work is proposed by Schmidt and Bannon (1992). They argue that people apply different heuristics and problem solving strategies which leave persistent traces in the information they take down. Accordingly, they claim, it is important to know and consider the originator of information in the process of interpreting the content of a common information space.
- Which *Action* was taken to resolve the problem. Besides, this field gives the support staff important information to consider when estimating the effort needed to accomplish the task. Considering the history of information has, furthermore, been suggested as an approach to reducing the risk of misinterpreting the content of a common information space (Kristoffersen 1995). Although applicable to all attributes of the record, we believe this is the most important for that purpose.
- *When* the problem was addressed is significant in terms of program versions, updates and configurations that are no longer relevant, or getting software off back-up tapes. Problems with PCs often appear to be due to installations of new applications, updates, etc. — operations often performed by the support staff.
- *Time-to-resolve* (found in the "Due" tapdown menu), which is the difference between date-of-commission and date-of-completion, could be an invaluable piece of information inasmuch as it helps members plan their working day. Clearly, this is an imprecise indication of the complexity of the problem, revealing more about the how busy the support person was, who responded to it than the demands of the tasks. Nevertheless, it can serve as a useful heuristic for planning.

Figure 2 shows how the support staff could benefit from the common experience space. In the example, a user has forgotten his password and the support

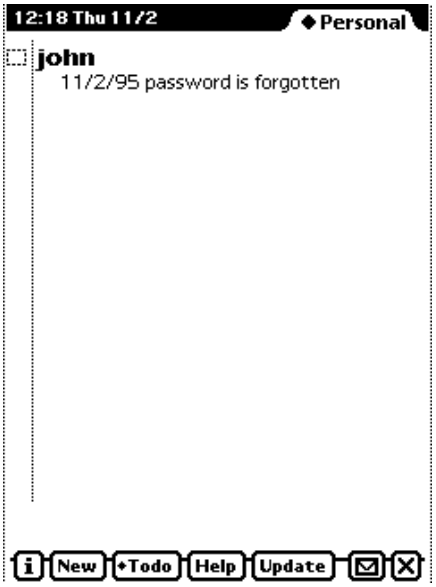
person has described the problem in terms of "forgotten password." By pressing the "Help" button, a search is made in the local experience space, using the "what" field as argument. The search resulted in one matching record: "password is forgotten," which was a task resolved by the person "john." When the support person taps a search result ("john"), the full record emerges.

FIGURE 2. Searching the common experience space; the figure shows that the problem "forgotten password" is used as an argument in the search of the local common experience space (First).
The result of the search, in this case "password is forgotten" is presented on the screen (Second).
By tapping the result, the full record emerges (Third).

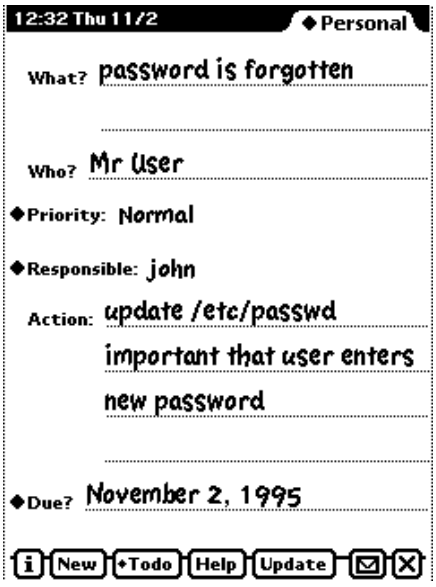
First:



Second:

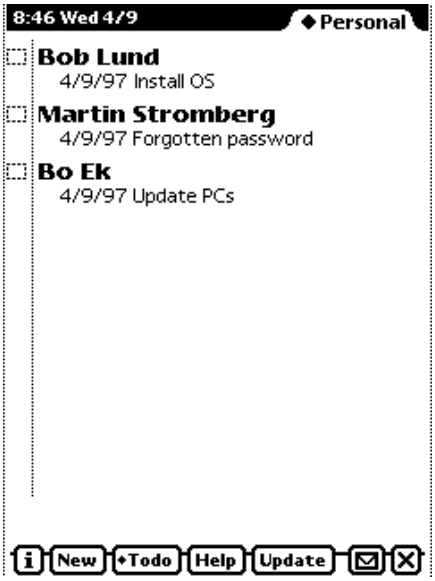


Third:



It could be useful for the group as a whole if entries which, upon inspection, were decided to be interrelated, could be linked to facilitate a group discussion, e.g., by imposing a newsgroup like structure to the common information space. This is not currently implemented in DARWIN, but will be considered in future versions of the application.

FIGURE 3. The individual to-do list



4.3. The individual to-do list

In the simplest mode of use, DARWIN is an electronic equivalent to the manual notepads already in use. However, browsing sequentially through all the entries in the local space would be inconvenient and time-consuming — much as navigation through the chronological journal on paper. Hence we wanted to implement an alternative view on the data, a personal to-do list. The small size of the screen makes it difficult to display the full records. The most important information for the user seems to be who did the work, what it concerned, and when it was done. Tapping on the record makes it open for full view.

Denying the user the possibility to change the fields in the list might encourage more elaborate record keeping in the data entry view, but only a practical, long-term evaluation will show whether this assumption is valid. The view is illustrated in Figure 3.

4.4. Coordinating tasks with others

When support members chose not to take the responsibility for a task themselves, they can negotiate it with others or redistribute it within the group. Transferring tasks to others can be done in two ways. When people meet each others physically, they can exchange tasks using the wireless beam feature provided by the Newton message pad computer. Connecting the Newton to a PC, people can exchange tasks with each other via the local area network. In the data entry operation the user sets the receiver of a task record (a specific person or the group), and upon pressing the “update” button, the task is transferred. The next time the “sender” connects her Newton to the PC, the tasks are transferred to the common task space where they are stored until the “receiver” connects again.

4.5. Coordinating tasks with the group

Moving commissioned requests into the common task space of the group should, in contrast to the person-to-person mode of sharing described above, be guided by rules implemented in the software. The reason why is that no-one yet has assumed responsibility for them. In the

first version of DARWIN, the *priority* given to a task by the originator determines whether it can be shared by the group at all, for how long it can lie idle in the common information space without someone picking it up, and who should be warned about it if nothing happens. We suggest the following functionality:

- Setting the responsibility to "group" (data entry operation) and pressing the "update" button, the task is marked for transfer to the common task space.
- When connecting to a workstation, the application automatically transfers the marked items to the common task space.
- When the PDA is connected to a PC, then the support person can press "group" in the "To-do" menu (button). This makes the group to-do list accessible and possible to browse.
- The support person can mark tasks in the group to-do list and transfer them to the individual to-do list on the PDA.
- When the PDA is connected, DARWIN updates the local and central common tasks spaces. The updated version of the task space is transferred to the PDA. This process can easily be terminated if this operation for some reason is not desirable.

There are two kinds of tasks in the common task space. There are not yet initiated tasks, i.e., tasks that have been entered into the information space but not yet checked out by anyone. The other kind are tasks that have been checked out but are not yet finished. Upon conclusion, tasks are not stored in the task space, but in the common experience space.

The issue was raised during interviews about whether high-priority commissions should be transferable. Our recommendation is that the person-to-person synchronous transfer should be allowed no matter what, because it has to be negotiated and accepted by the receiver. Aware of the risk of designing inflexible IT that people do not use, we suggest that social protocols (Ellis *et al.* 1991, p. 51) will deal with aspects of commitment appropriately.

4.6. A batch oriented approach

The work in the support group does not seem to require immediate, real-time update of the two common information spaces. Furthermore, such a solution is not technologically straightforward today, since access to the central information spaces would have to depend on mobile telephony. Even though mobile telephones are already in use in the group, such a solution would not be feasible for accessing the information spaces, because it is too cumbersome. The support personnel would, in order to access the information space, have to connect a mobile phone to the PDA, dial a number, and finally gain access to the spaces. We therefore chose a batch oriented approach for keeping, matching and updating information spaces locally (on the PDAs) and centrally on a server on the LAN. The information spaces update each others when the support staff connect their PDAs to the PC.

4.7. The flow of tasks

The members of the IT support group are normally approached by users whilst moving about in their workspace or common areas. As a result, unless the task can be solved immediately, they usually

enter the essential information about the query, to deal with it later. Sometimes other more important jobs, which thus are given a higher priority, are inserted into the queue of jobs. To organize and keep track of the tasks the support staff today use note books. DARWIN aims to replace the notebook by offering similar, but more sophisticated and flexible options for entering, organizing and keeping track of tasks. Using DARWIN, the support staff have five options when being approached by a customer asking for a service:

1. She can follow the user back to his workstation and get on with the job. If the support person believes the information about the task is vital for her, or the group, in the future, she can choose to insert a record by pressing the "New" button, describe the task, and chose "Done" option in the "Due" tapdown menu.
2. She can enter the task, by pressing the "New" button, promise to get back to the customer later, and place the task in the individual to-do list choosing the "myself" option in the "To-do" menu.
3. At this stage or later, she can tap the "help" button to see if there are other, similar requests already resolved by other members of the group, in which case she knows what to do. This might change the way this particular entry is articulated into the plan for the work.
4. At this stage or later, she can chose the "group" option in the "To-do" menu to submit the entry to the common task space. The task is transferred to the common task space the next time the PDA is connected to a PC.
5. If the support person wants someone else to do a task, e.g., because she lacks the competencies required, she first has to find the particular person. Typically the group meets for lunch once or twice a week, or they get in touch with each other over the phone. If her colleague agrees to take the job, the entry can be beamed across an infrared connection, or, when geographically dispersed, be transferred asynchronously via the common task space.

When someone assumes responsibility for the task, it appears in the personal to-do list. In some cases, the members elaborate on the problem description, e.g., if the user interpreted the situation wrongly, and if judged to be of common interest, the details about how the case was solved are entered. Completed records are transferred to the common experience space as soon as the message pad is connected to a PC.

The finished jobs are merged into the local space on each of the group member's PDA when they connect, hence becoming shared asynchronously. It is possible to update records using the PC. This is feasible, for example, when the actions taken to solve the problems could not comfortably be entered using the PDA. Tasks assigned to the group become accessible for the support person upon connecting the PDA to a PC. Tasks that have not been allocated until their expiration date are transferred back to the originator's PDA. Clearly, there will be a need for some database-administration in order to maintain a steady flow of work through the network. Ideally the

system should be transparent to the users, and maintain responsibilities that people have committed themselves to, for instance, by warning the originators if an entry with high priority is not taken on the first morning. We expect an empirical evaluation of the use of DARWIN to reveal a whole new set of requirements that take into consideration the social ordering of the workplace into which it is introduced.

5. Evaluation

In this section we present the evaluation of DARWIN, carried out in a group seminar and design briefings with individual users. The technical evaluation has been done in cooperation with the programmer involved in this project.

5.1. Group seminar

For the purpose of evaluating DARWIN, we organized a group evaluation seminar in which we presented and discussed the application and its use with the group members. The seminar lasted for two hours and was taped. The group concluded that they wanted us to develop the application for "real use."

Three main issues were brought to the fore in the overall discussion of using DARWIN in the group:

1. *The group acknowledged the potential of DARWIN to improve groupwork in the distributed, mobile and networked organization.* The support group as a whole acknowledged the suggested application as a feasible approach to improve the way they worked. They argued that the application seemed to sit well with the way they actually worked, acknowledging the potential to improve their group work in the distributed, mobile and networked organization.
2. *Computerized support that takes mobility into consideration*. During the seminar one group member told us that he had tested many different systems to organize work, both paper-based and computerized. None of them had, however, been appropriate, but rather caused extra work or had simply been impractical. Paper-based approaches offered little support for overall organization and co-ordination of work, he argued. He always ended up with "a million paper slips and PostIt notes of which it is impossible to get an overview." The computerized approaches, on the other hand, had consistently failed to address the mobile nature of his work forcing him to first take down information about a task on paper, before entering it into the computer. This member saw a great potential in overcoming these problems using a digital message pad as we had proposed.
3. *Messy support for "a messy reality."* The secretary of the group argued that DARWIN seemed "quite messy." She could not really concretize what appeared to her as messy, which perhaps, she argued, "is because I don't actually work with support." Group members involved more actively in the support work argued that DARWIN could not facilitate their work if it was not messy. They described their work in terms of many initiated tasks, a rapidly and unpredictably changing

work load, getting commissioned "everywhere," and so on: "I mean, the reality is messy for us, you know, it's the nature of our job," as one of them put it. Accordingly, DARWIN seems to meet the messy nature of the work in the group.

Another issue raised at the seminar was how the common task space should be used. The group concluded that it would not be desirable to enter tasks with high priority into the task space. The space should not accept tasks that required a detailed understanding of the customers' work. It would be more suitable for routine tasks like updating system versions and programming macros.[2] The issue of tracing the responsibility of tasks was also raised, and the group came to an agreement that the member given the task is responsible for it, until it was checked out from the task space. And although tasks should not be left in the task base forever — some kind of mechanism alerting the originator after a period of time was discussed — the application was deemed more suitable for commissions that were not time critical. Another important aspect discussed during the seminar was that the support staff should not be using the task space if they knew someone who could do this job well. A better alternative would be to use the mechanisms for direct co-ordination of work, i.e. the *others* view, or forfeit using DARWIN altogether.

Although the support group responded positively to our presentation of DARWIN, some critical factors were issued during the seminar. First, to establish a homogenous use of concepts to describe central aspects of the support work, it would be important to agree on which categories to be used and when. It was also argued that it would be very important to establish these categories *before* the support group take DARWIN in use — it would require much greater efforts to change the use of categories afterwards. Second, it was argued that it was important to establish the use rationale *before* introducing the application. According to some support group members, a lack of common understanding of the use partly explained why the previous experience data bases in the group had failed. Third, it would be essential that the support staff were trained in the use of DARWIN, as well as the Newton. Fourth, the application should not force the user to "write the same word five times before the machine understands it." The first version of the Newton operating system was criticized for its poor handwriting recognition. However, this problem seems to be considerably alleviated in a more recent version. Fifth, as we have argued above, DARWIN should be as easy to use as possible, and extravagant features should not be added to the first version of the application.

Supporting work with this class of IT applications is potentially interesting *beyond* the IT support group for this research organization, as many of its work groups and units are dispersed geographically.

5.2. Design briefings

Talking to the human factors specialist, a cognitive scientist employed by the group after we finished the first part of our investigation, revealed a small set of specific recommendations.

It should be possible to employ a fairly elaborate classification scheme for assigned tasks. The specialist suggested,

aligned with the IT support philosophy of the group's manager, that each item could be classified as *urgent* or not, and *important* or not. The result is a task topology illustrated in Table 1, with priorities in each entry.

TABLE 1. The task topology suggested in design briefing with human factors specialist

	Not important	*Important*
Urgent	**3**	**4**
Not urgent	**1**	**2**

Obviously, there is much to say about such a scheme, and we have chosen not to implement it in DARWIN. First, it is likely that the priorities of each task would be situation and person dependent as well. Second, an explicit representation of the table above would make the interface of the applications too awkward. We therefore decided to allow situated designation of priorities.

Furthermore, the specialist encouraged us to consider how to include more open task specifications, for instance by representing the network of people competencies. Thus, the users could get some help even if they did not know exactly how to specify the problem, for instance by getting directly in touch with colleagues. We think that this scenario is already addressed by DARWIN, since users are allowed to define problem categories of their own, to publish their special skills and competencies to the rest of the group.

The human factors specialist concluded that the notebook metaphor seemed logical and natural for its intended use. The layout is obviously different from the one that members of the group create on an ad-hoc basis, but she thought that "as soon as they get used to that it will be really good!"

5.3. Technical evaluation

The programmer of DARWIN finished the client prototype, and told us that the specifications were good. One serious problem was encountered, namely exchanging data between the Newton and PCs, but this can be explained by the immaturity of the platform rather than the conceptual design of the DARWIN application. In future versions, other PDAs and message pad computers will be considered, in particular with respect to the connection toolkits they make available to application programmers.

6. Discussion

Wagner (1994, p. 5) takes a perspective on the *networked organization* as fundamentally based on computer systems that offer the technical infrastructure for "connecting many actors and for accommodating the fluidity of people and spaces within and between organizations." The proposed application for co-ordination of work and sharing documentation seems to fit nicely into this conception. Wagner distinguishes between so-called *emergent* communication networks; denoting ensembles that continuously are established and dissolved on an ad-hoc basis, and *formal* networks. Emergent networks can be work groups (task centered), as in our example above, coalitions (political) or cliques (social). In our case-study organization, the network is clearly task-oriented with a set of support requirements to a technology that

can help them access the group functionality from a geographically distributed location.

For the purposes of design, the question of whether the network is a distinct social form is probably not essential. We depart from the notion that all organizations have some aspects of different *ideal types* in them (see, Thompson & Frances 1991). The question is rather: what can be elicited about the work, that is relevant to the use of an application such as the one outlined above, using the perspective on the organization as networked? Wagner claims that the network is typically loosely coupled, meaning that the ties between the actors are weak. This is certainly true in the decentralized organization of work in the research company. Few formal structures exist to initiate and control work activities and reward proper behavior (Kreiner & Schultz 1993). The level of activity is to some extent determined by the members of the IT support group themselves, and they are already planning and managing their working day on an individual level. Since the network is already based on commitment and trust (Ciborra 1993, p. 58), there is less danger of items getting stuck in the common task space than might be expected. A bureaucratic organization of the work is not a feasible alternative since it would reduce the potential of the members to carry out *proactive* support work.

Similarly, because of the loose coupling in a networked organization, people have different ways of accomplishing their work. It is paramount that applications intended to support networks do not *require* all of the members to use it all of the time. Grudin (1994, pp. 101) argues: "A word processor that is immediately liked by one in five prospective customers and disliked by the rest could be a big success. A groupware application to support five nurses that initially appeals to only one nurse in five is a big disaster." Supporting "networking" rather than group work only, DARWIN will only have to be used by *one* person, as a substitute for the manual notepad, in order to be useful since it affords more ways or organizing and navigating in the data. Our intention is, however, to create common information spaces as active resources for the whole group.

DARWIN does not unnecessarily bureaucratize the support network because each commission remains the responsibility of the member who agreed to do the job. Unless they chose to put the tasks into the open, common task space, it remains fully under their control. If it is submitted to the group, it becomes part of a more bureaucratic handling of the flow of work, required to stop items from disappearing. There is no increased control or reporting associated with this however. In the DARWIN design, items that go past their expiration date are simply returned to the sender, for renewed attention locally — an alert mechanism similar to the ones in the Coordinator (Winograd & Flores 1986). The person-to-person share involves face-to-face negotiation of the division of labor, and does not open the work item up to further bureaucratization.

The concerns raised by Orlikowski's (1992) well-known study of the use of a shared information space, are less relevant to this setting, we suspect, because of the nature of the workplace. The networked organization is based on mutual trust and there is little to be gained for the individual member by misinterpreting

information supplied by others (Ciborra 1993). After all, the option exists not to inspect or accept the work item. Competition is not a big issue, the positions available in the group are products of their physical localization and active "networking" with a set of users in the organization. The chances of IT support group members misunderstanding or not being able to use the information because it is interpreted out of context are less than in the case described by Orlikowski, because the tasks are less complex. Also, while they are dispersed in a network, the group members do know about each other's work and are, in a sense, *in context* through doing similar work themselves.

7. Conclusion

The point of departure for this paper was the empirical documentation of two problems that seem likely to be frequent in dispersed, mobile and networked groups: how to share experiences and coordinate work effectively. We have approached these problems by formulating the following research questions: *How could we facilitate sharing and coordination in the dispersed, mobile and networked group by means of IT?* Informed by the empirical work our suggestion proposed in the paper is DARWIN, a CSCW application supporting sharing and coordination in the dispersed, mobile and networked IT support group. The design of DARWIN was informed by close observations of the current work practices in the group. The evaluation of the application was positive, presumably due to continuous design briefings and close cooperation with the users.

In this paper, we have aimed to provide a rich illustration of how the general problem addressed could be resolved in a specific case. The results of the research presented here is the design, implementation, evaluation and discussion of the potential incorporation of the DARWIN application within the work of the IT support group. Based on these results it would be risky to draw general conclusions. Therefore, we view the research presented here as an initial investigation of CSCW technology use in mobile, dispersed and networking work settings. To make these ideas more generally applicable, more empirical work and design efforts are needed.

Further research concerns eliciting the general ideas of IT use for the mobile work settings. We will elaborate on the underlying architecture and protocol of DARWIN, in order to suggest a more general design framework for mobile CSCW applications. We will also experiment with wireless solutions and alternative interfaces to make further design suggestions that fit the specific conditions of the mobile worker.

Notes

[1]Computer-based Mechanisms of Interaction in Co-operative Work, Esprit Basic Research No 6225.

[2]An interesting comment inasmuch as we think programming macros for a user would benefit from a *proactive* approach.

Acknowledgments

This research is partly funded by the Swedish National Board for Industrial and Technical Development (NUTEK)

and the Swedish Work Environment Fund (AMFO). Support for this work was provided by the Research Council of Norway under project 100268/410, and Norsk Regnesentral. Thanks are due to the members of the IT support group. Bo Dahlbom, Urban Nuldén, Frank Reiff, Gareth Smith, Martin Strömber, Steve Viller, and Jacqui Forsyth are acknowledged for their help.

References

Ackerman, M., (1994). Augmenting organizational memory: A field study of Answer Garden, In *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, edited by R. Furuta and C. Neuwirth, pp. 243-252, Chapel Hill, NC: ACM Press.

Ackerman, M.S., and T.W. Malone, (1990). Answer Garden - A tool for growing organizational memory, In *Proceedings of The Conference On Office Information Systems*, edited by F. Lochovsky and R. Allen, pp. 31-39, Cambridge, MA.: ACM Press.

Allen, T., (1977). *Managing the flow of technology*, Cambridge, MA.: MIT Press.

Bannon, L., and S. Bødker, (1997). Constructing common information spaces, In *Proceedings of The Fifth European Conference on Computer-Supported Cooperative Work*, edited by J. Hughes *et al.*, pp. 81-96, Lancaster: Kluwer Academic Publishers.

Bannon, L., and K. Kuutti, (1996). Shifting perspectives on organizational memory: From storage to active remembering, In *Proceedings of Hawaii International Conference on System Sciences*, Hawaii: IEEE Computer Society Press.

Bellotti, V., and S. Bly, (1996). Walking away from the desktop computer: Distributed collaboration and mobility in a product design team, In *Proceedings of ACM 1996 Conference on Computer Supported Cooperative Work*, edited by K. Ehrlich and C. Schmandt, pp. 209-218, Cambridge, MA.: ACM Press.

Bowers, J., G. Button, and W. Sharrock, (1995). Workflow from within and without: Technology for cooperative work on the print industry shopfloor, In *Proceedings of The Fourth European Conference on Computer-Supported Cooperative Work*, edited by H. Marmolin, Y. Sundblad and K. Schmidt, pp. 51-66, Stockholm: Kluwer Academic Publishers.

Carstensen, P., C. Sørensen, and T. Tuikka, (1995). Let's talk about bugs!, *Scandinavian Journal of Information Systems*, 7(1): 33-54.

Ciborra, C., (1993). *Teams, markets and systems. Business innovation and information technology*, Cambridge: Cambridge University Press.

Conklin, J., (1993). Capturing organizational memory, in *Readings in Groupware and Computer-Supported Cooperative Work: Assisting human to human collaboration*, edited by R. Baecker, pp. 561-565, San Francisco, CA.: Morgan Kaufmann Publishers Inc.

Conklin, J., and M. Begeman, (1988). gIBIS: A hypertext tool for exploratory policy discussions, *ACM Transaction on Office Information Systems*, 6(4): 303-331.

Cook, S., and D. Yanow, (1993). Culture and organizational learning, *Journal of Management Inquiry*, 2(4): 373-390.

Ellis, C., S. Gibbs, and G. Rein, (1991). Groupware. Some issues and experiences, *Communications of the ACM*, 34(1): 39-58.

Grudin, J., (1994). Groupware and social dynamics: Eight challenges for developers, *Communications of the ACM*, 37(1): 93-105.

Hammersley, M., and P. Atkinson, (1993) *Ethnography. Principles and practice*, London: Routledge.

Holt, A., (1988). Displans: A new language for the study and implementation of coordination, *ACM Transactions on Office Information Systems*, 6(2): 109-125.

Joseph, A.D., J.A. Tauber, and M.F. Kaasoek, (1997). Mobile Computing with the Rover Toolkit, *IEEE Transactions on Computers*, 46(3): 337-352.

Karduck, A., (1994). TeamBuilder: A CSCW tool for identifying experts and team formation, *Computer Communications*, 17(11): 77-87.

Kim, D., (1993). The link between individual and organizational learning, *Sloan Management Review*, Fall, pp. 37-50.

Kistler, J., and A. Sayanarayanan, (1992). Disconnected operation in the Coda filesystem, *ACM Transactions on Computer Systems*, 10(1): 3-25.

Kreiner, K., and M. Schultz, (1993). Informal Collaboration in R & D. The Formation of Networks Across Organizations, *Organization Studies*, 14(2): 189-209.

Kristoffersen, S., (1995). Back to the future. Interfacing the history of the organisation, In *Proceedings of The 18th Information Systems Research Seminar in Scandinavia*, edited by B. Dahlbom *et al.*, pp. 359-375, Gjern, Denmark: Gothenburg Studies in Informatics.

Kristoffersen, S., and F. Ljungberg, (1996). Supporting mobility, co-ordination and sharing in dispersed, networking groups, In *Proceedings of The Fifth International Conference on Information Systems Development*, edited by S. Wrycza and J. Zupancic, pp. 339-354, Gdansk, Poland: University of Gdansk, Department of Information Systems.

Kristoffersen, S., and T. Rodden, (1996). Working by walking around. Requirements of flexible interaction management in video-supported collaborative work, In *Proceedings of Human Computer Interaction*, edited by B. Spence and R. Winder, London, UK: Springer-Verlag.

Langefors, B., (1995). *Essays on Infology. Summing up and planning for the future*, Lund: Studentlitteratur.

Ljungberg, F., (1997). *Networking*, Ph.D. thesis, Göteborg University, Göteborg, Sweden.

Malone, T., and K. Crowston, (1992). What is coordination theory and how can it help design cooperative work systems? In *Groupware: Software for computer-supported cooperative work*, edited by D. Macra and G. Bock, pp. 100-113, Los Almatios, CA: IEEE Computer Society Press.

March, J., (1991). Exploration and exploitation in organizational learning, *Organizational Science*, 2(1): 71-86.

Medina-Mora, R., T. Winograd, R. Flores, and F. Flores, (1992). The Action Workflow approach to workflow management technology, In *Proceedings of ACM 1992 Conference on Computer-Supported Cooperative Work*, edited by M. Mantei and R. Baecker, pp. 281-288, Toronto, Canada: ACM Press.

Migliarese, P., and E. Paolucci, (1995). Improved communications and collaborations among tasks induced by groupware, *Decision Support Systems*, 14(3): 237-250.

Munson, J.P., and P. Dewan, (1997). Sync: A Java framework for mobile collaborative applications, *IEEE Computer*, 30(6): 59-66.

Orlikowski, W.J., (1992). Learning from Notes: Organizational issues in groupware implementation, In *Proceedings of ACM 1992 Conference on Computer-Supported Cooperative Work*, edited by J. Turner and R. Kraut, pp. 362-369, New York: ACM Press.

Patton, M.Q., (1990). *Qualitative evaluation and research methods*, New York: Sage.

Pfeffer, J., (1982). *Organizations and organization theory*, Marshfield, MA.: Pitman.

Preece, J., (1994). *Human-computer interaction*, Wokingham, England: Addison-Wesley Publishing Company.

■

Sachs, P., (1995). Transforming work: Collaboration, learning and design, *Communications of the ACM*, 38(9): 36-44.

Schmidt, K., (1994a). Mechanisms of interaction reconsidered, in *Social Mechanisms of Interaction*, edited by K. Schmidt, pp. 15-122, Lancaster: ESPRIT BRA 6225 COMIC, Lancaster University, England.

Schmidt, K., (1994b). Modes and mechanism of interaction in cooperative work. Outline a conceptual framework, Risø National Laboratory, March 1994.

Schmidt, K., and L. Bannon, (1992). Taking CSCW seriously: Supporting articulation work., *Computer Supported Cooperative Work. An International Journal*, 1(1-2): 7-40.

Schmidt, K., and C. Simone, (1995). Mechanisms of Interaction: An Approach to CSCW Systems Design, In *Proceedings of International Workshop on the Design of Cooperative Systems*, edited by the COOP group, pp. 56-75, Antibes-Juan-les-Pins, France: INRIA.

Schmidt, K., and C. Simone, (1996). Coordination mechanisms: Towards a conceptual foundation of CSCW system design, *Computer Supported Cooperative Work: An International Journal*, 5(2-3): 155-200.

Stein, E., and V. Zwass, (1995). Actualizing organizational memory with information systems, *Information Systems Research*, 6(2): 85-117.

Svensson, L.G., and P. Orban, (1995). Inledning, in *Människan i tjänstesamhället*, edited by L. G. Svensson and P. Orban, pp. 11-23, Lund: Studentlitteratur.

Sørensen, C., P. Carstensen, and H. Borstrøm, (1994). We can't go on meeting like this! Artifacts making it easier to work together in manufacturing, In *Proceedings of The Australian Conference on Computer-Human Interaction*, edited by S. Howard and Y. Leung, pp. 181-186, Melbourne, Australia: ACM Press.

Terry, D.B., (1995). Managing update conflicts in Bayou, a weakly connected replicated storage system, In *Proceedings of 15th ACM Symposium on Operating Systems Principles*, edited by J. Bennett and M. Weiser, pp. 172-183, New York: ACM Press.

Thompson, G., and J. Frances (ed.), (1991). *Market, hierarchies & networks. The coordination of social life*, London: Sage.

Wagner, I., (1994). Networking actors and organisations, *Computer Supported Cooperative Work. An International Journal*., 2(1-2): 5-20.

Winograd, T., and F. Flores, (1986). *Understanding computers and cognition. A new foundation for design*, New Jersey: Ablex Publishing Corporation.

Yakemovic, K.C.B., and E.J. Conklin, (1990). Report on a development project use of an issue-based information system, In *Proceedings of ACM 1990 Conference on Computer-Supported Cooperative Work*, edited by T. K. Bikson, Los Angeles, California: ACM Press.

■

■